

SOFTWARE STATUS @ LMD

ROMAN KLASSEN

RUHR UNIVERSITY BOCHUM

10.10.2022



- Collection of C++ Apps for compute-heavy calculations
- And Python scripts to organize the multiple apps

Working principle:

- Measure angular distribution of $\bar{p}p$ -scattering at low angles
- Compare with DPM Model
- Calculate Luminosity from comparison

Find the code at:

<https://github.com/panda-luminosity-detector-group/LuminosityFit>

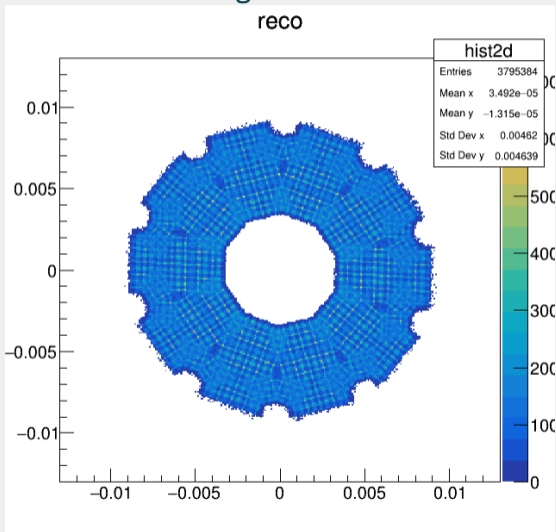
WHAT MAKES THIS SO DIFFICULT?

- Reduced acceptance (between sensors, primary beam pipe)
- Limited (albeit high) resolution
- Large data sample required

Solution:

- Generate comparison data with reduced acceptance & resolution as well
- Both must model the real detector *exactly*
- Process multiple smaller data samples concurrently

Reconstructed angular data:



WHY DO WE NEED SO MUCH CODE?

Luminosity fit process once reconstructed data is there:

1. With Lumi_QA data:
 - ▶ Determine IP position
 - ▶ Calculate total cross section via DPM (no external data required)
2. Then, these can be done simultaneously:
 - ▶ With IP position, apply XY cut on Lumi_QA data (that means new reconstruction run, hundreds of data files)
 - ▶ Calculate acceptance and resolution by:
 - Generating 10^8 MC events (Box or DPM)
 - full reconstruction incl. cuts
 - determine acceptance & resolution from reconstructed data
3. Calculate Luminosity:
 - ▶ Merge data packets from hundreds of reconstruction jobs
 - ▶ fit measured data to simulated reconstructed data (only three free variables, but 2D model and *lots* of data)

Each step is done by C++ app(s), and the overall recipe is automated with python.

Working*

Working*

*But some code refactoring is still required.

PANDAROOT

DATA SIMULATION AND RECONSTRUCTION

Where does Lumi_QA data come from?

For simulations:

- MC gen
 - ▶ Box gen
 - ▶ DPM gen
- Digitization

→ Sensor data

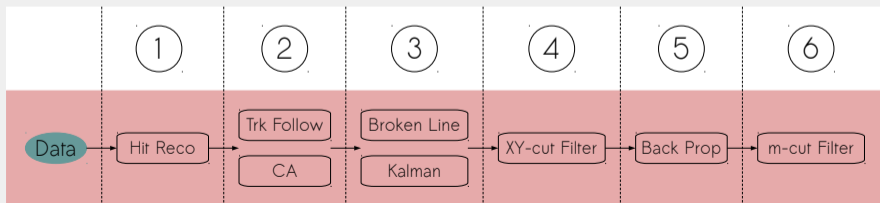
Always:

- Hit Reconstruction
- Track Search/Fitter
- Back Propagation
- QA

→ reconstructed tracks

And lastly:

- Alignment
- Lumi Fit



SIMULATION MACROS

Currently, each step must be done individually with a ROOT macro:

- `root -l -q runPixeloSimBox.C(param1, param2, ...)`
- `root -l -q runPixel1Digi.C(param1, param2, ...)`
- `root -l -q runPixel2Reco.C(param1, param2, ...)`
- `root -l -q runPixel2a...`
- `root -l -q runPixel2b...`
- and so many more

Automation: A bash script that asks for some run parameters and then runs those in succession. Also, the Luminosity Fit Software runs those macros.

Most of them use FairRunSim and FairRunAna, but switching to PndMasterTask is on the agenda

CONTAINERIZATION

You want to run Luminosity Fit

Requires PandaRoot ↓

PandaRoot: built upon ↓

FairRoot: requires ↓

FairSoft: provides ↓

External Packages (ROOT etc.)

And of course some installed packages.

Which means for a new user, all these packages have to be downloaded and compiled.

On Workstation and on cluster.

THE GOAL: CONTAINERIZATION

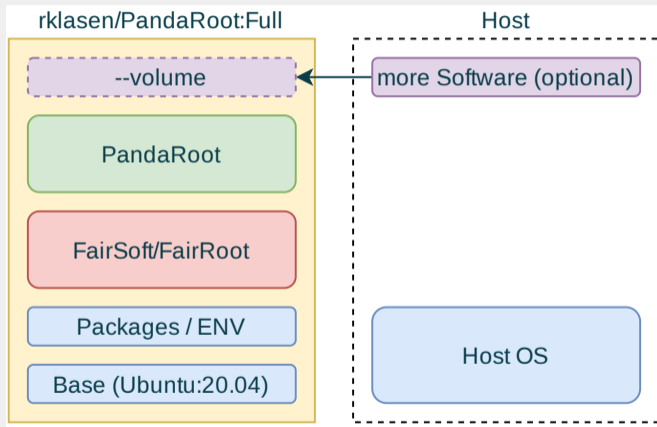
Use PandaRoot on any machine (with docker or Singularity) without installing FairSoft/FairRoot. Avoid dependency/compiler hell.

- MC Data Generation on HPC
- Analysis on Workstation
- Software Development

Advantage: easy setup and clean separation from host system.

POSSIBLE LAYOUTS 1: ALL-INCLUSIVE

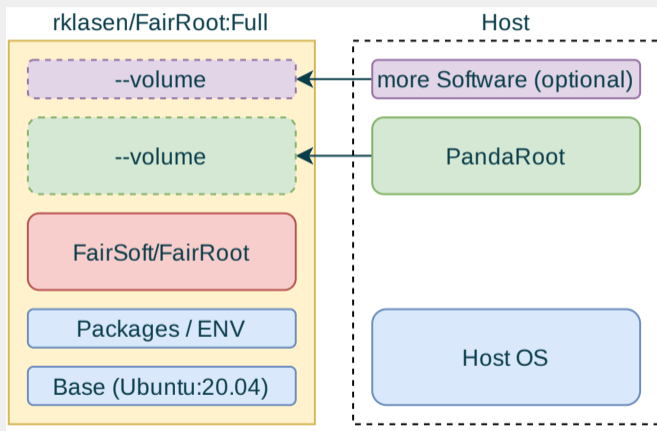
Complete PandaRoot installation with all necessary packages, ready to go.



Con: Large Image Size (May run into issues on HPC systems)

POSSIBLE LAYOUTS 2: FAIRSOFT/FAIRROOT FULL

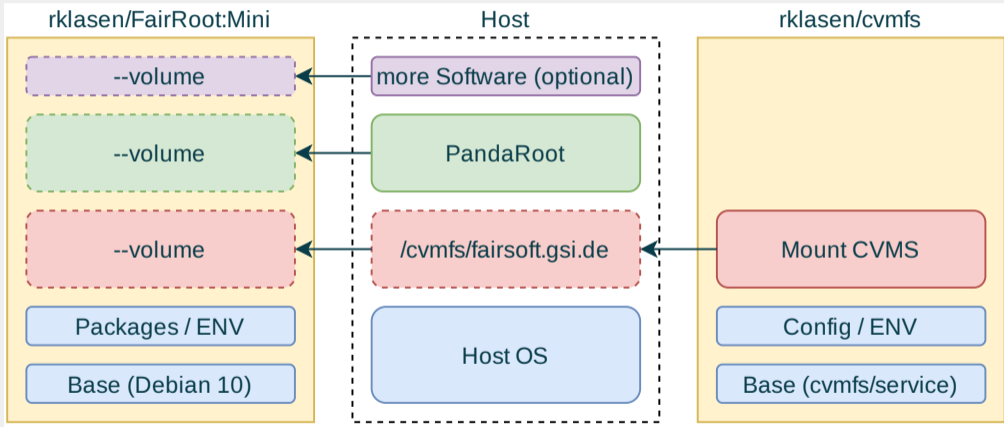
Includes FairSoft/FairRoot but needs external PandaRoot source directory.



Good for code development, this is what I'm using right now.

POSSIBLE LAYOUTS 3: EXTERNAL EVERYTHING

Sandbox ready for PandaRoot, but doesn't include FairRoot/FairSoft or PandaRoot

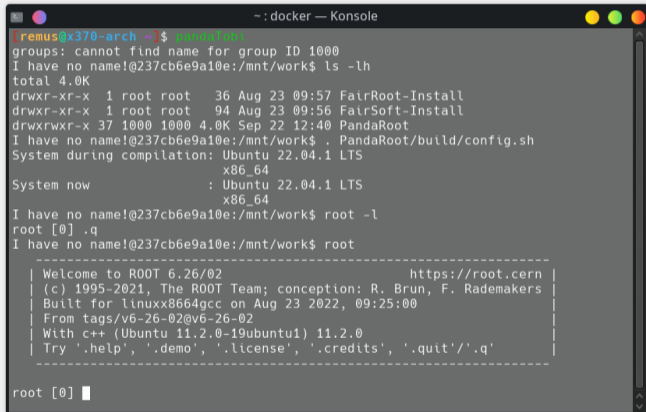


Pro: Use auto builds from GSI. **Bonus:** Mount CVMFS via another container!

PANDAROOT IN CONTAINER

When onboarding a new user (Workstation):

- Clone Luminosity Fit and PandaRoot repos
- Pull container image
- Mount Luminosity Fit and PandaRoot repos to container and run



```
remus@x370-arch ~$ panda1cbl
groups: cannot find name for group ID 1000
I have no name!@237cb6e9a10e:/mnt/work$ ls -lh
total 4.0K
drwxr-xr-x  1 root root   36 Aug 23 09:57 FairRoot-Install
drwxr-xr-x  1 root root   94 Aug 23 09:56 FairSoft-Install
drwxrwxr-x 37 1000 1000 4.0K Sep 22 12:40 PandaRoot
I have no name!@237cb6e9a10e:/mnt/work$ . PandaRoot/build/config.sh
System during compilation: Ubuntu 22.04.1 LTS
                           x86_64
System now                  : Ubuntu 22.04.1 LTS
                           x86_64
I have no name!@237cb6e9a10e:/mnt/work$ root -l
root [0] .q
I have no name!@237cb6e9a10e:/mnt/work$ root

-----
| Welcome to ROOT 6.26/02                                     https://root.cern
| (c) 1995-2021, The ROOT Team; conception: R. Brun, F. Rademakers
| Built for linuxx8664gcc on Aug 23 2022, 09:25:00
| From tags/v6-26-02@v6-26-02
| With c++ (Ubuntu 11.2.0-19ubuntu1) 11.2.0
| Try '.help', '.demo', '.license', '.credits', '.quit'/'.q'
-----

root [0] █
```


- Luminosity Fit Software is working on Himster
- Adaption to VRIGO in progress
- Some code refactoring still required
- Containerization is well documented and works for simulations and analysis
- Alignment software needs to be integrated into new Luminosity Fit code

THANK YOU FOR YOUR ATTENTION!